

## Software Architecture Design

### Architecture Evaluation

Duc-Man Nguyen, Ph.D  
[mannd@duytan.edu.vn](mailto:mannd@duytan.edu.vn)  
0904 235 945

### Lesson Objectives

- During this session we will discuss general software architecture evaluation concepts
- We will introduce an SEI evaluation method
  - Architecture Tradeoff Analysis Method (ATAM)

## Is This a Good Architecture?



## Good Architecture?

- Architectures aren't inherently good or bad
- They may or may not be "fit for purpose"
- As part of the design process it makes sense to periodically review the architecture with respect to the purpose

## When To Evaluate

- Some organizations have an “official” architecture review as a milestone in their process
  - Often part of a “phase gate”
- There are many times when it makes sense to evaluate, however
  - When considering an acquisition
  - When the organizational strategy shifts
  - Prior to the production phase
  - As part of the design process

## Evaluation as Part of the Design Process

- Design is a hierarchical process
- It makes sense to periodically evaluate where you stand with respect to the goals
- ACDM recommends doing this after each portion of significant design activity
  - Evaluation occurs in stage 4 in ACDM
- This works well with iterative life cycle models



## Ways to Evaluate Architectures – 1

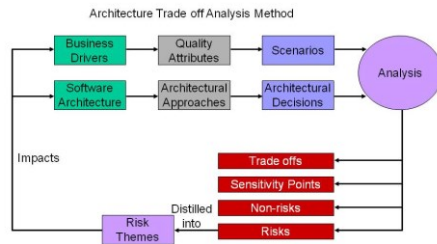
- There are various approaches to evaluating software architectures
  - simulations
    - tool based
    - generally aids understanding dynamic aspects
  - formal model verification
    - architectures are formally modeled and the model checked or verified
    - generally focuses on proving the existence or absence of various properties

## Ways to Evaluate Architectures – 2

- There are various approaches to evaluating software architectures (continued)
  - prototype
    - build a facsimile
    - refine the architecture based on the experience
  - scenario based evaluation
    - quality attribute and use case scenarios are used to guide the evaluation of the software architecture
    - has the “feel” of a structured design walk-through

## The ATAM - 1

- The Software Engineering Institute has developed the Architecture Tradeoff Analysis Method (ATAM).



## The ATAM - 2

- The purpose of the ATAM is *to assess the consequences of architectural decisions in light of quality attribute requirements and business goals*.
- ATAM is a stakeholder centric approach that assumes you don't have a set of prioritized quality attribute scenarios
- Keep in mind who developed ATAM and what the purpose was

## Purpose of the ATAM – 1

- The ATAM is a method is a scenario based evaluation method
  - The method assumes that the organization hasn't explicitly identified or codified quality attribute requirements.
  - Scenarios are developed by stakeholders and used to analyze the architecture to discover risks.
  - Risks uncovered during the ATAM evaluation can then be made the focus of mitigation activities such as further design, further analysis, prototyping, research, and so forth.
  - Surfaced tradeoffs, non-risks, and sensitivity points can be explicitly identified and documented.

## Purpose of the ATAM – 2

- The purpose of ATAM is **NOT**
  - to fix, prioritize, or determine the impact of risks during ATAM – *this is a post ATAM activity*.
  - to provide precise, formal analyses.
- The purpose **IS**
  - to find general *trends* – correlations between architectural decisions and system properties.
  - to discover any risks, sensitivity points, tradeoffs, and non-risks created by architectural decisions.

## When to Use the ATAM – 1

- The ATAM can be used throughout the lifecycle – *provided there is a software architecture to evaluate.*
- The ATAM can be used
  - after an architecture has been specified but there is little or no code
  - to evaluate architectural alternatives
  - to evaluate the architecture of an existing system

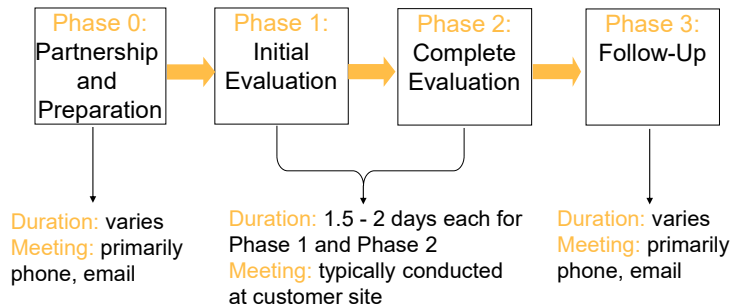


## When to Use the ATAM – 2

- To perform an ATAM evaluation, *there must be a software architecture to evaluate.*
- What does it mean to "have a software architecture?"
  - Representations of the "to-be-built" or "as-built" system are required for an ATAM evaluation.
  - Prose is available that describes the architecture.
  - An ATAM evaluation is inappropriate if the software architecture of the system has not been created (or discovered) yet.

## Nominal ATAM Schedule

ATAM evaluations are conducted in four phases.



## ATAM Phase 0 – 1

- **Phase 0** precedes the technical part of the evaluation.
  - The customer and a subset of the evaluation team exchange understanding about the method and the system whose architecture is to be evaluated.
  - A go/no-go decision is made by the evaluation leader.
  - If the ATAM is a go, an agreement to perform the evaluation is worked out.
  - A core evaluation team is fielded.
  - The evaluation team begins reading the architectural documentation provided.



## ATAM Phase 0 – 2

- The purpose of the go/no-go decision is to ensure the success of the ATAM evaluation.
  - Evaluators must be able to understand the state of the architecture well enough to make a decision and ensure that the candidate system is ready for evaluation.
  - If a no-go decision is made, we explain to our clients the reasons for declining, and suggest remediation steps, or work with our customers to help with develop the architecture – this is NOT part of ATAM.

## ATAM Phase 0 – 3

- Key considerations for the go/no-go decision.
  - A responsible architect has been identified and is committed to participate in the ATAM evaluation.
  - A business manager, program manager, and/or system sponsor has been identified and is committed to participate in the ATAM evaluation.

### ATAM Phase 0 – 4

- Architectural documentation has been provided to the evaluation team and is sufficient enough to proceed with the ATAM evaluation.
  - There is a context drawing and multiple representations of the system (module, C&C, allocation).
  - There is sufficient prose to describe the system.



### ATAM Phase 0 – 5

- The business drivers and architectural presentation has been provided to the ATAM evaluation team.
- The presentations are sufficient, but well within 1.5 hour time constraints for each presentation during the ATAM evaluation.
- The appropriate stakeholders have been identified and are committed to participate in the ATAM evaluation.
- The list of participants and their roles with respect to the system has been sent to the ATAM evaluation team.

## The Evaluation Team – 1

- ATAM Evaluation teams vary in size from 4 to 6 people.
- Each ATAM evaluation team consists of an evaluation leader and at least three additional members. Roles include:
  - Facilitator — facilitates discussions, brainstorming, and analysis.
  - Scribe — writes utility tree, raw scenarios, risks, sensitivities, and tradeoffs on ATAM electronic templates, a second scribe my transcribe information on flipcharts.

## The Evaluation Team – 2

- Each ATAM evaluation team consists of an evaluation leader and at least three additional members. Roles include: (continued)
  - Process observer — monitors the process steps and takes notes about the process, and how it could be improved.
  - Timekeeper — informs the moderator/facilitator when the time allocated for a step has expired.
  - Questioner(s) — raise issues that the stakeholders have not thought of; asks questions based on how the quality attributes of interest relate to architectural styles.

## ATAM Phase 1

- **Phase 1** involves a small group of predominantly technically oriented stakeholders.
- Phase 1 is
  - architecture-centric
  - focused on eliciting detailed architectural information and analyzing it
  - top-down analysis



## ATAM Phase 1 Steps

1. Present the ATAM
2. Present business drivers
3. Present architecture
4. Identify architectural approaches
5. Generate quality attribute utility tree
6. Analyze architectural approaches
7. Brainstorm and prioritize scenarios
8. Analyze architectural approaches
9. Present results

**Phase 1**

## Step 1: Present the ATAM

- The evaluation team presents an overview of the ATAM
  - rationale for evaluating software architectures
  - steps in brief
  - techniques
  - outputs

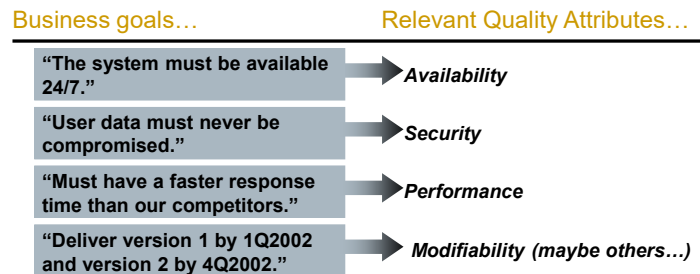
## Step 2: Present Business Drivers – 1

- An ATAM customer representative describes the system's business drivers including the
  - business context for the system
  - high-level functional requirements
  - high-level quality attribute requirements
    - architectural drivers: the quality attributes that “shape” the architecture
    - critical requirements: the quality attributes that are most central to the system's success

## Step 2: Present Business Drivers – 2

Quality attributes are derived from business goals.

Evaluators MUST LISTEN CAREFULLY... the quality attributes wiz by very quickly...



## Step 3: Present Architecture

- The architect presents an overview of the architecture including the
  - technical constraints such as an OS, hardware, or middleware prescribed for use
  - other systems with which the system must interact
  - architectural approaches used to address quality attribute requirements

### Step 4: Identify Architectural Approaches

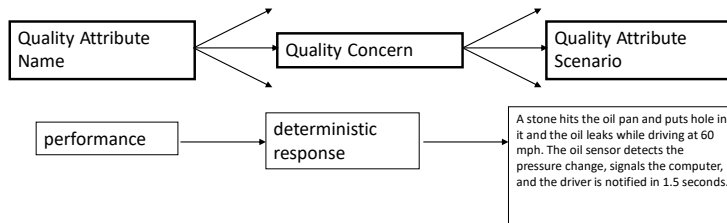
- Distill the quality attributes and associate with business drivers
- Distill the architectural drivers.
- They will identify predominant architectural approaches and the reason they were selected by the architect.
- The evaluators will then identify tactics, patterns, and other mechanisms in the architecture that are key to realizing quality attribute goals – and therefore the business drivers.

### Step 5: Generate Quality Attribute Utility Tree – 1

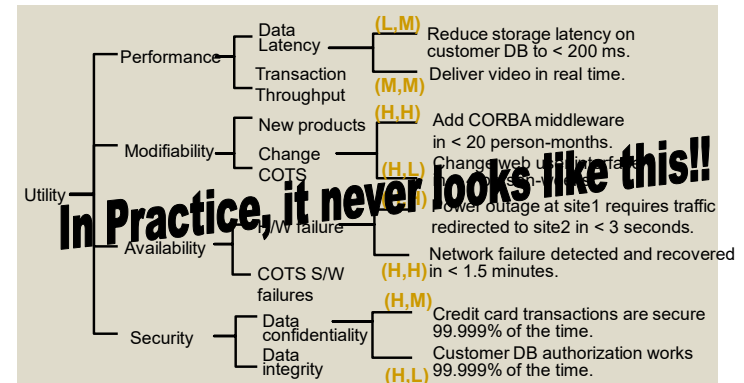
- ATAM assumes that the client has never formally identified their quality attributes.
- Identify, prioritize, and refine the most important quality attribute goals by building a *utility tree*.
  - A utility tree is a top-down vehicle for characterizing the “driving” attribute-specific requirements.
  - The most important quality goals are the high-level nodes (typically performance, modifiability, security, and availability).
  - Scenarios are the leaves of the utility tree.

### Step 5: Generate Quality Attribute Utility Tree – 2

- The utility tree organizes, characterizes, and prioritizes the quality attribute requirements.
- The utility tree is what makes phase one “top down.”
- The utility tree has three major parts:



### Step 5: Generate Quality Attribute Utility Tree – 3





## Step 5: Generate Quality Attribute Utility Tree – 4

Phase I: Quality Attribute Utility Tree		
Quality Attribute	I. List the quality attribute such as security, modifiability, etc.	
Attribute Concerns	A. This is the first quality attribute concern for the Top Level Quality Attribute	
Scenarios	1. This is a scenario that illuminates this quality attribute concern	(I, D)
	1. :	:
	1. This is a scenario that illuminates this quality attribute concern	(I, D)
Attribute Concerns	B. This is another quality attribute concern for the Top Level Quality attribute.	
Scenarios	1. This is a scenario that illuminates this quality attribute concern	(I, D)
	1. :	:
	1. This is a scenario that illuminates this quality attribute concern	(I, D)
Attribute Concerns	...	
Scenarios	1.:	(I, D)
	1.:	:
	1.:	(I, D)

## Step 5: Scenarios

- Scenarios are used to
  - represent stakeholders' interests
  - understand quality attribute requirements
- Scenarios should cover a range of
  - anticipated uses of the system
  - anticipated changes to the system
  - unanticipated stresses on the system
- A good quality attribute scenario
  - clearly states the stimulus and the responses of interest.
  - is very specific

## Stimuli, Environment, Responses

- Use case scenario  
*The remote user requests a database report via the Web during peak period and receives it within 5 seconds.*
- Growth scenario  
*Add a new data server to reduce latency in scenario 1 to 2.5 seconds within 1 person-week.*
- Exploratory scenario  
*Half of the servers go down during normal operation without affecting the overall system availability.*

## Step 6: Analyze Architectural Approaches – 1

- Prior to analysis the raw stakeholder scenario is expanded into a full 6 part scenario.
  - source, stimulus, environment, artifact, response, and response measure
- The analysis begins by asking the architect:

*“Given the stimulus, from the source, affecting the artifact, under the described environmental conditions; show how the architecture responds within the response measure indicated by the scenario.”*

## ATAM Analysis Template

ATAM Phase <###> Scenario <###>	
<b>Scenario</b>	Cut and paste scenario here. In phase 1, scenarios will be cut-and-paste from the Utility Tree Template. In phase 2, scenarios will be cut-and-paste from the Phase 2 Scenario Template.
<b>Business Goal(s)</b>	List those business goals that this scenario effects.
<b>Attribute</b>	Applicable in phase 1 - cut and paste from Utility Tree Template.
<b>Attribute Concern</b>	Applicable in phase 1 - cut and paste from Utility Tree Template.
<b>Scenario Refinement</b>	<b>Stimulus</b> Condition affecting the system/artifact
	<b>Stimulus Source</b> Entity generating stimulus
	<b>Environment</b> Conditions under which stimulus occurred
	<b>Artifact</b> Part of system stimulated (can be whole system)
	<b>Response</b> Activity undertaken after arrival of stimulus
	<b>Response Measure</b> Testable measure of the activity taken after the arrival of the stimulus
<b>Architectural Decisions and Reasoning</b>	List the architectural decisions relevant to this scenario that affect quality attribute response and a discussion of the qualitative and/or quantitative rationale for why the architectural decisions contribute to meeting the quality attribute response requirement.
<b>Risks</b>	List risks any discovered.
<b>Sensitivities</b>	List sensitivities any discovered.
<b>Tradeoffs</b>	List tradeoffs any discovered.
<b>Non-risks</b>	List non-risks any discovered.
<b>Other Issues</b>	List issues any discovered.

## Step 6: Analyze Architectural Approaches – 2

- The architect uses the architectural representations to step through the scenario to describe how the architecture responds to the stimulus.
- ATAM evaluators and stakeholders can ask probing questions to uncover
  - risks
  - non-risks
  - sensitivity points
  - tradeoffs

## Risks, Non-Risks Sensitivity Points, and Tradeoff

- A **risk** is a potentially problematic architectural decision.
- **Non-risks** are good architectural decisions that are frequently implicit in the architecture.
- A **sensitivity point** is a property of one or more components (and/or component relationships) that is critical for achieving a particular quality attribute response.
- A **tradeoff** is a property that affects more than one attribute and is a sensitivity point for more than one attribute.

## Risks and Tradeoffs

- Example risk:
  - *“Rules for writing business logic modules in the second tier of your three-tier architecture are not clearly articulated. This could result in replication of functionality thereby compromising modifiability of the third tier.”*
- Example tradeoff:
  - *“Changing the level of encryption could have a significant impact on both security and performance.”*

## Sensitivity Points and Non-Risks

- Example sensitivity point:
  - *“The average number of person-days of effort it takes to maintain a system might be sensitive to the degree of encapsulation of its communication protocols and file formats.”*
- Example non-risk:
  - *“Assuming message-arrival rates of once per second, a processing time of less than 30 ms, and the existence of one higher priority process, a one second soft deadline seems reasonable.”*

## Exercise

- Can we skip any step in phase 1? Explain why.
- Name some alternative method.

## ATAM Phase 2

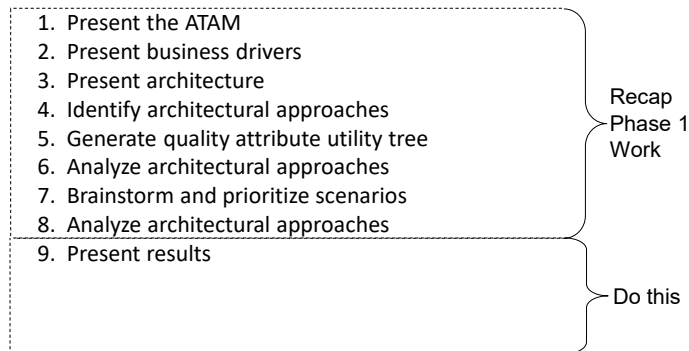
- Phase 2 involves a larger group of stakeholders and is
  - focused on eliciting diverse stakeholders points of view and on verification of the Phase 1 results
  - stakeholder-centric
  - bottom-up analysis



## Hiatus Between Phase 1 and Phase 2

- During the hiatus between phase 1 and phase 2, the evaluation team will
  - consolidate their notes, the utility tree, risks, non-risks, sensitivity points, and tradeoffs
  - build a presentation for phase 2 that recapitulates the results of phase 1
  - formulate potential risk themes
  - begin creating the final presentation and writing the final report
  - plan the logistics for phase 2

## ATAM Phase 2 Steps



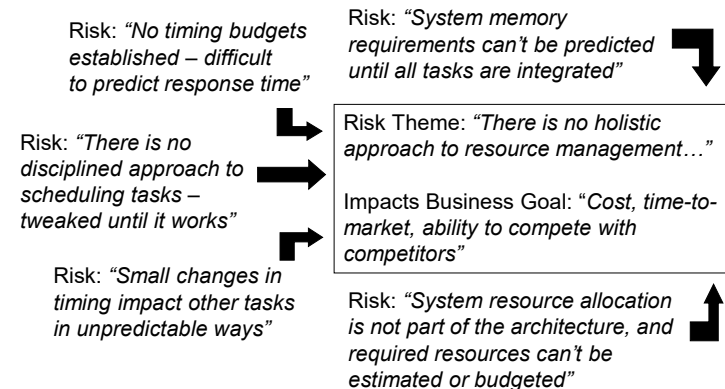
## Step 7: Brainstorm and Prioritize Scenarios

- The group of stakeholders is larger and more diverse than those present in Phase 1.
  - may include more than just techies
  - you will have more stakeholders present
- Stakeholders generate scenarios using a facilitated brainstorming process.
  - Scenarios at the leaves of the utility tree serve as examples to facilitate this step.
- In Phase 2, each stakeholder is allocated a number of votes roughly equal to  $0.3 \times \text{\#scenarios generated}$ .

## Step 8: Analyze Architectural Approaches

- This step continues the analysis of step 6 with the larger group of stakeholders using the new scenarios.
  - Additional risks and non-risks are identified.
  - Additional architectural information is annotated as it is uncovered.
- At the conclusion the evaluation team will call for a long break (2 hours or so)
  - During the break we will complete the final presentation.

## Risk Themes





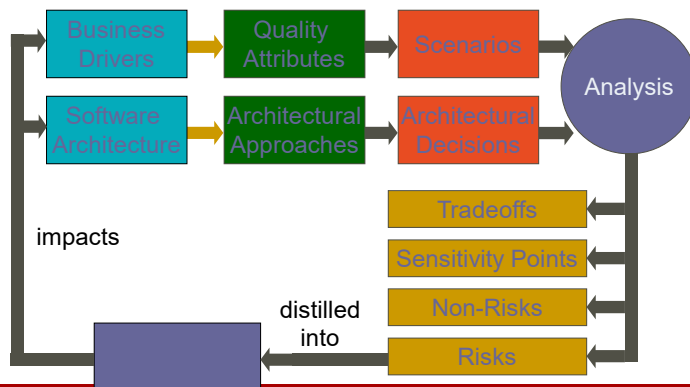
## Step 9: Present Results

- Typically the evaluation leader makes the final presentation.
- There is never time to cover each slide in detail, so we focus on high-level details.
  - the ATAM process, duration, participants
  - overview of the business goals
  - overview of the system architecture
  - the utility tree.
  - summary of analysis
  - summary of risks, sensitivities, non-risks, tradeoffs
  - risk themes

## ATAM Phase 3

- **Phase 3** primarily involves producing a final report for the customer, as well as assessing the quality of the evaluation and the ATAM materials.

## Conceptual Flow of the ATAM



## Issues With ATAM – 1

- While ATAM can be very useful, some issues with ATAM include:
  - The ATAM is disruptive – it requires that you stop everything, prepare, evaluate, and fix.
  - Despite claims to the contrary, ATAM seems to fit best in BIG waterfall lifecycles – with BIG budgets.
  - An ATAM evaluation is onerous and expensive – as a result small projects and organizations are reluctant to adapt it.



## Issues With ATAM – 2

- The quality of the ATAM analysis (steps 6 & 8) is dependent upon the training, intuition, and experience of the evaluators.
- ATAM doesn't say anything about the achievement of functional requirements
- Still, no tool support.

## Issues With ATAM – 3

- ATAM does not tell you what to do with the output of the evaluation and there is no formal process for what to do with ATAM output.
  - This is partially due to the nature of the SEI business model.
  - Has made it difficult to make it a transitionable, self-sustaining method (again, this is in tension with the SEI business model).
- It requires a significant infrastructure to adapt, utilize, and maintain ATAM as a practice within an organization.

## Continuous Evaluation

- A better approach to designing and evaluating architecture might be to continually evaluate as the architecture matures rather disruptively evaluate.
  - treat design as a technical process, not as a mere milestone or artifact
  - weave architecture into the design, production, and maintenance processes
  - use the architecture as the anchor for the entire project to develop plans, mitigate risks, align work force and so forth

## Tailoring The ATAM

- There are many aspects of the ATAM that could be tailored to suit a given context
- If for example you are articulating and prioritizing architecture drivers throughout you don't need to do so in the ATAM
  - This eliminates the need to have the “stakeholder centric” portions of the method
  - It could also eliminate the need to generate and prioritize scenarios
- It could also make sense to include reviews of the functional drivers

## Session Summary

- Software architecture evaluation is an effective way to discover the consequences of architectural decisions
  - before detailed design and implementation
  - major resources are committed to upgrade a system
- We introduced an SEI evaluation method ATAM

## Group Work

- What are the pros and the cons of ATAM?
- Can we use ATAM for all kinds of projects?



## References - 1

- Bass, L.; Clements, P. & Kazman, R. *Software Architecture in Practice, Second Edition*. Boston, MA: Addison-Wesley, 2003.
- Clements, P.; Kazman, R. & Klein M. *Evaluating Software Architecture: Methods and Case Studies*. Reading, MA: Addison-Wesley, 2002.
- Lattanze, Anthony *Architecting Software Intensive Systems* Boca Raton, FL: Auerbach, 2009

## References - 2

- <https://www.youtube.com/watch?v=xR-7fjAoORI> (Evaluation ATAM)